Fundamentals of SOA

Presented to: SOA Brown Bag #1

By:

Sam Ceccola – SOA Architect
703 980 4012
sam.ceccola@doveltech.com

Date:

Agenda

- **What is SOA?**
- **What is a Service Oriented Enterprise?**
- **Basic SOA Architecture**
- **SOA Concepts**
  - Definition of a service in SOA
  - XML in SOA
  - Service registries / repositories
  - Web services in SOA
  - ESB in SOA
  - SOA Governance
  - Synchronous and Asynchronous Execution
- **Challenges of implementing SOA**

What is SOA:

- SOA is an architecture approach for defining, linking, and integrating reusable business services that are self-contained with their own functionalities. The services are independent of application or IT infrastructure platform

- SOA takes IT to another level, one that's more suited for interoperability and heterogeneous environments. You can interconnect between enterprises regardless of their supported infrastructure, which opens doors to delegation, sharing and reuse of existing services

What is a Service Oriented Enterprise ?

A Service Oriented Enterprise (SOE) is an organization whose business processes and IT infrastructure are integrated across the entire enterprise to deliver on-demand services to customers, partners and suppliers

Migrating into a SOA can greatly enhance an enterprise's agility, but evolving into a SOE goes beyond the promise of SOA. To become a SOE, an organization needs to:

- "Experience the service"
- Drive architecture solutions directly from the business processes
- Be model-driven
- Enable architectural transparency (e.g., what, where, why, cost, relationships, etc.)
- Treat rules, policies and processes as "first class citizens"
- Automate Service Governance

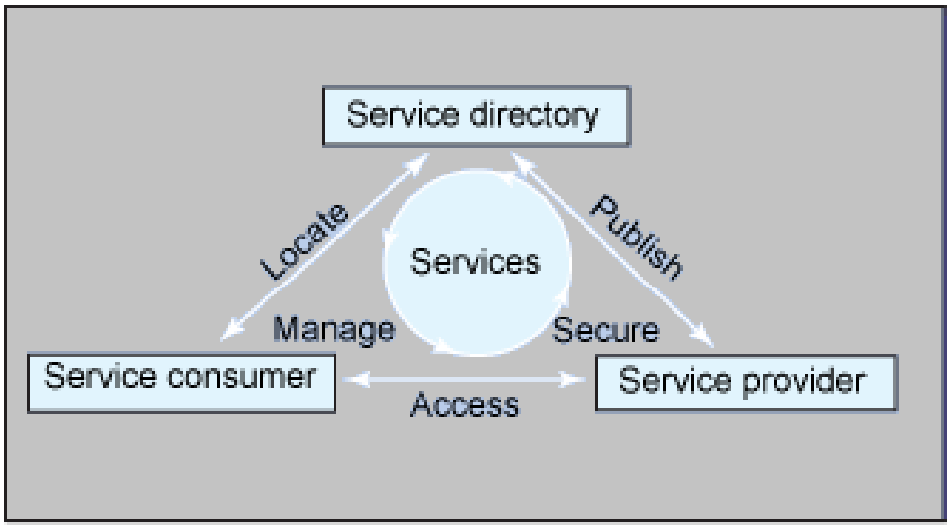
SOA Key Characteristics

- **Technology Neutral**
- **Standards Based**
- **Loosely coupled**
- **Process- and Service-centric**
- **Composability**
- **Layering**

Basic SOA Architecture

A basic SOA architecture is composed of a service provider, service, and a service directory

First the service provider creates a service and decides to expose it and publish it. Publishing is done by posting the service information on the service directory. On the other hand, a service requester in need of a certain service,searches the service directory for one that meets the necessary criteria



SOA Concepts

- **Definition of a service in SOA**
- **XML in SOA**
- **Service registries**
- **Web services in SOA**
- **ESB in SOA**
- **SOA governance**

SOA Concepts: Definition of a service in SOA

- A service is a function that is well-defined, self-contained, and does not depend on the context or state of other services
- Loosely coupled services are independent of other services
- These Services are registered in a directory service so that they can be located during design or runtime
- Service definition is not dependent on transport protocol used to access it
- Service can be used in a location/platform/technology agnostic approach

SOA Concepts: XML in SOA

- XML is the foundation for virtually all web services standards, such as

  XML schema, SOAP, WSDL and UDDI
- Using XML resolves the challenge of working with different data formats in different applications across multiple platforms

SOA Concepts: Service Registries / Repositories

A service registry is a directory of services available in an SOA system. It contains the physical location of services, versions and validity periods of services, service documentation, and policies. A service registry is one of the main building blocks of an SOA architecture

- **Design time use**

  Developer searches through the registry for existing services that can take part in an orchestrated application

  Interest is in the functionality but not the specific instances

- **Run time use**

  Application searches through the registry for the service but focus is now on implementations of it

SOA Concepts: Web Services in SOA

- **Web services are the cornerstone of SOA**

– Enforce standards and thus, promote compatibility and portability
– Are cross-platform and cross-language
– Are widely supported, making SOA relatively easy to adopt
– Are message-oriented.

SOA Concepts: ESB in SOA

- **An ESB supports an event based integration platform for SOA**
  - Routing services are created for each event to be supported
  - Routing services receive the event from a native system and will route it to one or multiple destinations
  - Destinations can be additional services within the ESB or BPEL processes or external services
  - Content based routing is supported through rules definition
  - Data transformations may be required before routing to a destination
  - Routes can be specified to be synchronous or Asynchronous

SOA Concepts: SOA Governance

- Without a controlling entity, SOA is not only challenging to manage, but it invites chaos because of its open and distributed nature. Because of this, it needs a management and controlling entity

– Governance assures keeping services at a defined level in terms of integrity, performance, reliability, and currency.
– Governance also enforces security of enterprise data and privacy of information shared across boundaries
– It enforces integrity and reliability of data
– It ensures a certain level of performance and quality of all services

SOA Concepts: Synchronous and Asynchronous Execution

- **Synchronous and Asynchronous Execution**
  - Services can be defined to be synchronous i.e. a response will be provided back to the client after the event handling is completed
  - Services can be defined to be asynchronous i.e. a response will be provided back to the client as a separate invocation on the client

Challenges of Implementing SOA

- **Lack of documented architecture:**
  - **One of the key components of SOA is the architecture, however, many organizations' infrastructures lack a clearly documented architecture**
- **Lack of control:**
  - **Program Managers feel that they can't control the various services with respect to quality, schedules, changes, budget and therefore they elect to duplicate the functionalities in that specific silo to ensure that the totality of the solution can be controlled and delivered as planned**
  - **Security:**
    - **Since a solution needs to make use of multiple services, how can the security be guaranteed? How can PM feel comfortable that security is addressed?**
  - **Service level:**
    - **A final cause for concern to PM is the service level; since PM is responsible for delivery of a solution to end-users with specific up-time requirement reliance on multiple services with "unknown" up-time can be a bit scary.**